Natural Language Processing Readiness Reporting Enhancement

Final Report

prepared for the

Logistics Integration Agency, US ARMY

in fulfillment of subcontract

USA-TPSU-HRB-0393-1373

to the

Pennsylvania State University

by the

Center for Intelligent Information Processing
A Joint Research Facility of:

HRB System, Inc.
300 Science Park Road
State College, PA 16804

and

The Pennsylvania State University
College of Engineering
University Park, PA 16802

1 Introduction

The Center for Intelligent Information Processing (CIIP), a joint research facility of HRB Systems and the Pennsylvania State University, is pleased to submit this final report to the Logistics Integration Agency, US ARMY.

1.1 Background

The ASORTS database provides ODCSLOG with invaluable information regarding the readiness of the almost 5,000 reporting US ARMY units. Each unit sends in a unit readiness status report on a monthly basis. These reports are converted into machine readable form and loaded into the ASORTS database for analysis. The purpose of this analysis is to provide monitoring for possible future problems that may arise within or across units. The main area of investigation centers around equipment based unit problems. Along with certain objective data, unit commanders can also include subjective analysis in the form of commander's comments. The objective data provides much information that is useful but the commander's comments provide answers to "why" questions. Unfortunately, at the present staffing level it is impossible to read all the commander's comments. Indeed, we estimate that, given the volume of material to read, it would take an above average reader almost 33 hours just to read through the comments at a high rate of speed. The purpose of this project has been to determine the feasibility of using Natural Language Processing technology to enhance analysis of the gold mine of information contained in commander's comments.

1.2 Report Outline

This final report is divided into two sections. The first section introduces a possible solution to the problem of extracting important information from commander's comments in readiness reports. We call this system the Comments Database. It is from this baseline that enhancements may be made. The second section gives a description of three proposed enhancements that may be made to the Comments Database. We give a brief description of each along with an estimate of the feasibility of the work as well as the scope of the work needed to implement the enhancement if it is indeed feasible.

2 Comments Database

2.1 Concept of Operations

The Comments Database is meant to be an effective tool for aiding in the analysis of commander's comments in ASORTS readiness reports. The idea is to allow the analyst to search through a database containing the important concepts found in the commander's comments. By a concept we mean the thing that stays the same when you say the same thing in different ways. For example **A-1** and **Abrams** both refer to a particular model of tank. Thus, they refer to the same concept.

The advantage of this approach can be illustrated with the following scenario borrowed from the experience of a former unit commander. This particular commander downgraded his unit's readiness status from C1 to C2 in one instance. The reason was that he noticed a deterioration of the treads of his unit's tanks. This was noted in his readiness comments. An ASORTS analyst took note of this and spent many hours searching through the other unit commander's comments to find out if others were having this trouble. As it turned out, other commanders had indeed noted the same problem and a disaster was averted. Given a conceptual query such as ***tank treads with problem OR problems with tank treads,*** a conceptual search engine would be able to find most of the other commander's comments with regard to tank tread problems. In fact, the phrases "tank tread problems" and "a deterioration of the treads of his unit's tanks" as found in this report both refer to this concept and would likely have been found. Given this powerful capability, the creative analyst should be able to extend his or her effectiveness in the investigation of equipment based problems as reported in commander's comments.

2.2 Requirements

2.2.1 System Specification

Equipment The system will require a Windows NT machine running at least 200MHz and possessing at least one gigabyte of disc space.

User Interface All processes and procedures should be controlled through a user interface, including the following:

- **On line Help** On line help should be available for every process and procedure available to the user.

- **Querying Concept Database** The system should step the user through the process of forming, submitting, saving and retrieving queries. The system should step the user through the process of viewing and analyzing results.

- **Indexing Concept Database** The system should step the user through the process of indexing a new set of commander's comments and notify the user when the process is done. The user should be able to stop the indexing process at any time and restart it from where it left off or from the beginning after clearing the concept database.

- **Display of System Status** The status of the system should be viewable by the user. The system should step the user through the process of monitoring and interpreting the system status.

- **System Maintenance** The system should step the user through all processes related to maintaining the system, including system set up, system start up, database clearing, system restart, system stop, and maintenance of the NLP engine. The system should step the user through the process of specifying or modifying the properties of the NLP engine as well as the definition of what concepts are to be extracted during the indexing phase.

Indexer The system should have an indexer which queries the main ASORTS database for the comments of commanders of each unit, extracts concepts from those comments and stores that information in a manner that is retrievable by both unit and conceptual content.

- **Query of Main ASORTS Database** A query should be formed for each unit in succession from the main ASORTS database. This assumes that unit designation is a key of this database. The commander's comments should be separated and passed on to the NLP engine for conceptual analysis.

- **NLP Engine** The NLP engine should analyze each sentence of every commander's comments, record the important concepts contained in each sentence and pass them back to the indexer to be associated with a unit and stored in the concept database.

- **Concept Storage** The indexer should associate each unit with the important concepts extracted from the commander's comments and store them in a concept database.

Concept Database The unit and concept information should be stored and be retrievable from a concept database which has the following storage and retrieval requirements.

- **Storage** Each unit should be associated in the database with its concepts and each concept should be associated with the unit of commander's comments from which it came. Enough information should be associated with each unit to allow access to the original comments found in the main ASORTS database.

- **Retrieval** The comment database should allow retrieval of information either by reference to unit identifier or through a conceptual description of the comments made by the unit commander. For example, a reference to equipment should retrieve all references to that equipment whether referred to by LIN, NSN, model or description.

2.2.2 Installation and Maintenance

The contractor should install the system and should make provision for maintenance of the system.

2.3 High Level Design

In this subsection we describe at a high level a possible design of the Comments Database that satisfies the requirements we have outlined above.

2.3.1 Prerequisites

In order for such a system to be created, tested and installed, it is paramount that a copy of the ASORTS binaries and source code be provided as well as a copy of an old Bernoulli disc with a month's worth of readiness reports. The absence of any of this information renders the creation of the Comments Database unfeasible.

2.3.2 Overall Architecture

The overall architecture is illustrated in Figure 1 at the back of this report. The dotted line represents the boundary between existing ASORTS database code and the code generated for the Comments Database. By way of the user interface, the user commands the indexer to read and index all units' commander's comments. After indexing, the user can then query the concept database either by reference to units or description of the comments made by their commanders.

2.3.3 User Interface

The user interface has three main functions. The first is to run the indexer, the second is to query the concept database and the third is to monitor and maintain the system. This third function includes the specification of the lexicon, grammar, text structure, concept mappings and concept hierarchies, as well as the definition of what concepts are to be extracted during the indexing phase. All existing code would have to be ported to Windows NT.

2.3.4 Indexer

The indexer has three main functions. The first is to query the main ASORTS database by unit for each unit's commander's comments. The second is to extract conceptual information, by way of an NLP engine, from these comments, the third is to associate the concepts extracted with the unit and store this information in the concept database. The second of these functions requires a sophisticated NLP engine capable of recognizing and extracting important concepts from commander's comments. HRB System's Cognitive Processing Engine (COPE) meets these requirements exactly. COPE is the heart of the ConceptCrawler web search tool which searchs the World Wide

Web for web pages of interest with very high precision (98%) and recall (over 80%) according to recent tests. All existing code would have to be ported to Windows NT.

2.3.5 Concept Database

The concept database has only one function, to store and associate unit identifiers and concepts contained in units' commander's comments and to retrieve that information efficiently. Currently this is done in the ConceptCrawler with a database developed jointly by HRB Systems and the Pennsylvania State University. To our knowledge, no other database has this capability. All existing code would have to be ported to Windows NT.

2.4 Scope

The overall scope of building the Comments Database is estimated in this section. This estimate is based on the analysis that follows.

2.5 Assumptions

All estimates are made by considering lines of existing C++ code. Wherever extensive changes are to be made, we assume that the code must be completely rewritten. New code can be written at the rate of 12 lines per (eight hour) day. Existing code that will not require extensive changes we assume to be conversion code. Conversion code can be written at a rate of 60 lines per day. Reused code running under the same platform is called "test only code." This can be tested at the rate of about 120 lines per day. Given the number of lines per day we can estimate the number of man years required based on a rate of 260 days per man year.

2.5.1 User Interface

The portion of the COPE engine which compiles specification files (a subcomponent of the user interface) has 8373 lines of C++ conversion code. The rest of the interface will require extensive changes and therefore is considered to be 1,700 lines of new code. Finally, the specification files are expected to take about one man year to update. These consist of the lexicon and hierarchy files which run in the hundred of thousands of lines of specifications. The specification files need to include words and hierarchies for place names (a gazeteer), equipment names (by LIN and NSN as well as common names), personnel designation by rank and MOS, as well as 30 to 40 thousand commonly used words. The common words are relatively easy to derive from our present lexicon. We do not have a gazeteer so one would have to be sought and perhaps purchased. Information concerning equipment and personnel is available through LIA. The only other consideration in creating the specifications is to relax the English grammar to allow parsing of Military style English which has its own set of quirks and colloquialisms. This gives a total of two man years.

2.5.2 Indexer

The present indexer of the ConceptCrawler consists of control code and the NLP engine portion of COPE. The control code can be expected to change radically. It consists of 3,369 lines of C++ code. The NLP engine, consisting of 5,864 lines, can be converted. This gives a total of about two man years.

2.5.3 Concept Database

The present concept database of the ConceptCrawler will require extensive changes since it is written for the SHORE database instead of SYBASE, the DBMS used in ASORTS. We expect the lines of code to increase by 50% since SHORE is an object database and SYBASE is purely relational. The lines of code in the present concept database is 4361 lines. Therefore we expect the new line count to be about 6500. This gives a total of two man years.

2.5.4 Conclusion

It is clear that the cost is high. However, this is primarily due to the necessity to convert from a UNIX platform to an NT platform. If that were already accomplished prior to starting the program then we estimate that the project would only require about three man years since all the conversion costs would be eliminated. The conversion itself is estimated to require about three man years. We give a recommendation in the final section how to break this work up into two separate pieces.

3 Enhancements

In this section we consider three possible enhancements to the Comments Database, each of which we will evaluate by determining first if it is a feasible enhancement and, if so, what is its scope.

3.1 Integrated ASORTS Database

The first enhancement that should be made is to integrate both the Comments Database together in a single application with the existing ASORTS database code. Without this step it will be impossible to implement advanced database searching techniques such as relating objective data with commander's comments. It is also far more convenient to have just one application running rather than switching between two.

Given HRB System's vast software and database experience we estimate a cost of about two man years over a performance period of six months. The success of this endeavor depends on the successful procurement of all existing government owned source code.

3.2 Upgrade to COE Compliance

Another enhancement that can be made is to upgrade the system to DII-COE compliance so that the integrated ASORTS and Comments databases can be used on the SPRNet. The COE requirements appear to be exceedingly complex. The documentation available online consists of about 500 pages of detailed specifications. We cannot make a credible estimate as to the scope at this time without further study.

3.3 Enhanced Searching and Data Mining

In this last section we will assess the feasibility of using advanced techniques for readiness analysis using commander's comments.

3.3.1 Statistics

Statistical queries will usually take one of the following three forms.

What percentage of units have property P?

What percentage of units with property P have property Q?

Therefore, the gathering of statistics requires first the ability to detect properties of units within the database and commander's comments. For example, having a personnel rating of P2 is a property of units as well as having downgraded to C2 status. These properties are directly accessible from the objective data. Properties such as having a particular reason for downgrading (for example, personnel shortages) are found only in commander's comments. As long as references to these reasons are straight forward (that is, not stated in an obscure manner) they can be detected and put in the comments database for statistical analysis. Thus, it is entirely feasible for an analyst to make a query such as "what percentage of those downgrading readiness to C2 give personnel shortages as a reason?" However, this presupposes that the ASORTS and Comments databases are integrated into a single application.

3.3.2 Explanatory Linkage to Objective Data

Analyzing text to determine what it refers to is a very difficult task. However, simple references can be detected and properly matched. For example, the following comment was made by a commander regarding unit personnel strength.

THE NUMERIC PERSONNEL STATUS OF THE UNIT IS P1 HOWEVER, ...

The commander has made a reference to the personnel status entry in the database. This is easily detected by the direct reference to personnel status or by the reference to a P1 rating. Other references can be much more obscure and therefore very difficult to detect accurately. The reason for this is that fairly heavy duty inferencing is required as well as encyclopedic knowledge of military operations in order to detect indirect references to database fields. This fact does not negate, however, the potential usefulness of detecting direct references which we determine to be quite feasible.

3.3.3 Cause and Effect Analysis

Analyzing cause and effect is an advanced research problem that is not currently feasible to build into the ASORTS database. Too little is known and the techniques available are not well understood or standardized. We therefore do not suggest that LIA look into implementing this capability at this time.

3.3.4 Conclusions

Assuming that the ASORTS and Comments databases have been successfully integrated and that references within commander's comments are not put in an overly obscure or oblique manner, we determine that it is entirely feasible to gather useful statistics from both objective data and commander's comments. We also conclude that given direct references we can effectively link explanations within commander's comments to objective data within the ASORTS database. Concerning DII-COE compliance, we find that it is premature to make a determination at this time without further study. Finally, we conclude that cause and effect analysis is not feasible at this time.

4 Recommendations

Based on our analysis we make the following recommendations:

- Support a conversion to NT in the form of a demonstration system that is similar to the Web based demo we currently have. This would require about three man years over a six month performance period.
- Support a more complete study of the scope of conforming to the DII-COE requirements specification. This would require about one half a man year over a three month performance period.
- Support the production of a Comments Database that would compliment the ASORTS database (also useful to run in tandem with the ARMS database). This would require about three man years over a six month performance period.
- Support the integration of the ASORTS and Comments databases into a single application. This would require about two man years over a six month performance period.

- Support the enhancement of the integrated ASORTS database with the capabilities suggested in the previous section. We do not have an estimate of the scope of this endeavor at this time.

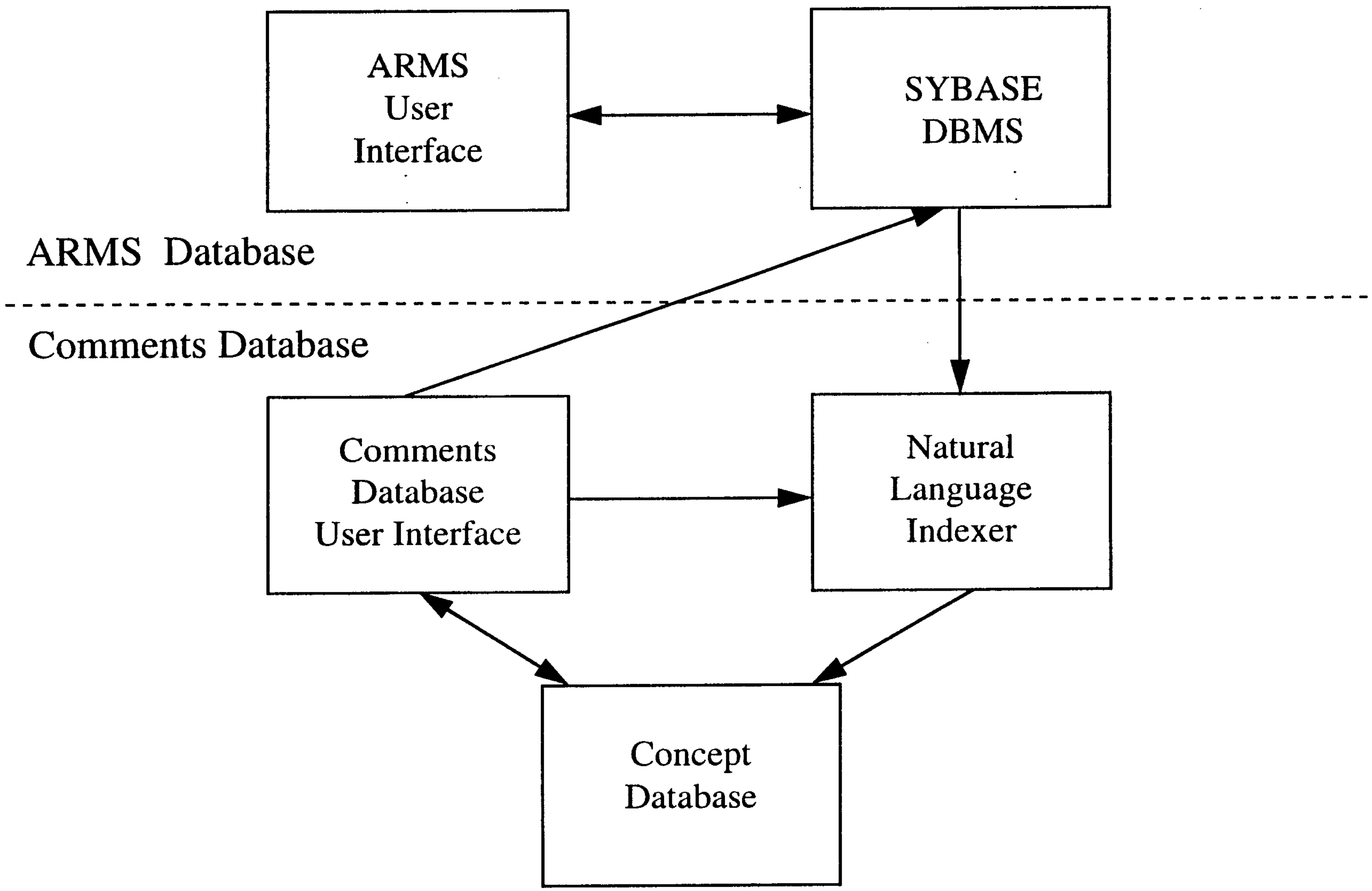

Figure 1. High Level Design of the Comment Database

REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

Public Reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comment regarding this burden estimates or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188,) Washington, DC 20503.

1. AGENCY USE ONLY (Leave Blank)

2. REPORT DATE
Feb. 28,2000

3. REPORT TYPE AND DATES COVERED
Final Report 9/23/97-9/22/98

4. TITLE AND SUBTITLE
Multiagents in Logistics Environment

5. FUNDING NUMBERS
DAAG55-97-1-0393

6. AUTHOR(S)
Soundar R.T. Kumara

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
The Pennsylvania State University,
University Park, PA 16802

8. PERFORMING ORGANIZATION REPORT NUMBER
USA-TPSU-HRB-0393-1373

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U. S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
ARO 37724.1-MA

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12 a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12 b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

In this research, a Natural Language Processing System to enhance readiness reporting is developed. This work is needed for logistics, as the email messages sent between different participants needs to be understood and related to logistics. A concept database and the indexing and querying is developed. Three possible enhancements to the existing comments database are proposed. A practical system is delivered to theUSA Logistics Integration Agency. This work is jointly conducted by HRB Systems Inc., and the Pennsylvania State University.

14. SUBJECT TERMS

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OR REPORT
UNCLASSIFIED

18. SECURITY CLASSIFICATION ON THIS PAGE
UNCLASSIFIED

19. SECURITY CLASSIFICATION OF ABSTRACT
UNCLASSIFIED

20. LIMITATION OF ABSTRACT
UL

Enclosure 1